F 13

```
MMM         MMM TTTTTTTTTTTTTTT HHH         HHH RRRRRRRRRRRR    TTTTTTTTTTTTTTT LLL
MMM         MMM TTTTTTTTTTTTTTT HHH         HHH RRRRRRRRRRRR    TTTTTTTTTTTTTTT LLL
MMM         MMM TTTTTTTTTTTTTTT HHH         HHH RRRRRRRRRRRR    TTTTTTTTTTTTTTT LLL
MMMMMM   MMMMMM       TTT       HHH         HHH RRR         RRR       TTT       LLL
MMMMMM   MMMMMM       TTT       HHH         HHH RRR         RRR       TTT       LLL
MMMMMM   MMMMMM       TTT       HHH         HHH RRR         RRR       TTT       LLL
MMM   MMM   MMM       TTT       HHH         HHH RRR         RRR       TTT       LLL
MMM   MMM   MMM       TTT       HHH         HHH RRR         RRR       TTT       LLL
MMM   MMM   MMM       TTT       HHH         HHH RRR         RRR       TTT       LLL
MMM         MMM       TTT       HHHHHHHHHHHHHHH RRRRRRRRRRRR          TTT       LLL
MMM         MMM       TTT       HHHHHHHHHHHHHHH RRRRRRRRRRRR          TTT       LLL
MMM         MMM       TTT       HHHHHHHHHHHHHHH RRRRRRRRRRRR          TTT       LLL
MMM         MMM       TTT       HHH         HHH RRR   RRR             TTT       LLL
MMM         MMM       TTT       HHH         HHH RRR   RRR             TTT       LLL
MMM         MMM       TTT       HHH         HHH RRR   RRR             TTT       LLL
MMM         MMM       TTT       HHH         HHH RRR      RRR          TTT       LLL
MMM         MMM       TTT       HHH         HHH RRR      RRR          TTT       LLL
MMM         MMM       TTT       HHH         HHH RRR      RRR          TTT       LLL
MMM         MMM       TTT       HHH         HHH RRR         RRR       TTT       LLLLLLLLLLLLLLL
MMM         MMM       TTT       HHH         HHH RRR         RRR       TTT       LLLLLLLLLLLLLLL
MMM         MMM       TTT       HHH         HHH RRR         RRR       TTT       LLLLLLLLLLLLLLL
```

```
**FILE**ID**MTHBITOPS                                   J 13



MM      MM  TTTTTTTTTT  HH      HH  BBBBBBBB      IIIIII    TTTTTTTTTT    OOOOOO    PPPPPPPP      SSSSSSSS
MM      MM  TTTTTTTTTT  HH      HH  BBBBBBBB      IIIIII    TTTTTTTTTT    OOOOOO    PPPPPPPP      SSSSSSSS
MMMM  MMMM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP      PP  SS
MMMM  MMMM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP      PP  SS
MM  MM  MM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP      PP  SS
MM  MM  MM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP      PP  SS
MM      MM      TT      HHHHHHHHHH  BBBBBBBB        II          TT      OO      OO  PPPPPPPP        SSSSSS
MM      MM      TT      HHHHHHHHHH  BBBBBBBB        II          TT      OO      OO  PPPPPPPP        SSSSSS
MM      MM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP                  SS
MM      MM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP                  SS
MM      MM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP                  SS      ....
MM      MM      TT      HH      HH  BB      BB      II          TT      OO      OO  PP                  SS      ....
MM      MM      TT      HH      HH  BBBBBBBB      IIIIII        TT        OOOOOO    PP          SSSSSSSS        ....
MM      MM      TT      HH      HH  BBBBBBBB      IIIIII        TT        OOOOOO    PP          SSSSSSSS        ....


LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

Table of contents

```
    (2)        50       HISTORY          ; Detailed Current Edit History
    (3)        67       DECLARATIONS
    (4)        98       MTH$IIAND
    (5)       136       MTH$JIAND
    (6)       175       MTH$IIOR
    (7)       213       MTH$JIOR
    (8)       251       MTH$IIEOR
    (9)       289       MTH$JIEOR
    (10)      327       MTH$INOT
    (11)      365       MTH$JNOT
    (12)      403       MTH$IISHFT
    (13)      442       MTH$JISHFT
```

```
0000          .TITLE  MTH$BITOPS         ROUTINES FOR BIT OPERATIONS
0000          .IDENT  /1-005/            ; File: MTHBITOPS.MAR  Edit: JAW1005
0000
0000  ;
0000  ;****************************************************************************
0000  ;*                                                                          *
0000  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000  ;*  ALL RIGHTS RESERVED.                                                    *
0000  ;*                                                                          *
0000  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0000  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0000  ;*  TRANSFERRED.                                                            *
0000  ;*                                                                          *
0000  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0000  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000  ;*  CORPORATION.                                                            *
0000  ;*                                                                          *
0000  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0000  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000  ;*                                                                          *
0000  ;*                                                                          *
0000  ;****************************************************************************
0000  ;
0000  ;
0000  ; FACILITY: MATH LIBRARY
0000  ;++
0000  ; ABSTRACT:
0000  ;       This module contains routines for operations on individual
0000  ;       bits of arguments.
0000  ;
0000  ;
0000  ;--
0000  ;
0000  ; VERSION: 0
0000  ;
0000  ; HISTORY:
0000  ;
0000  ; AUTHOR:
0000  ;       Jonathan M. Taylor, 14-JUL-77: Version 0
0000  ;
0000  ; MODIFIED BY:
0000  ;
0000  ;
0000  ;
```

```
0000              .SBTTL  HISTORY          ; Detailed Current Edit History
0000
0000
0000      ; Edit History for Version 0 of MTH$BITOPS
0000      ;
0000      ; 1-001 - Update version number and copyright notice.  The last
0000      ; 1-002 - Add "_" to the PSECT directive.  JBS 21-DEC-78
0000      ;           edit number for version 0 was 3.  JBS 16-NOV-78
0000      ; 1-003 - Fix MTH$IIAND, MTH$JIAND.  SPR 11-25743  SBL 8-Aug-1979
0000      ; 1-004 - Add routines to support additional functions defined in
0000      ;         MIL-STD 1753 for Fortran: MTH$IMVBITS, MTH$JMVBITS,
0000      ;         MTH$IIBITS, MTH$JIBITS, MTH$IISHFTC, MTH$JISHFTC, MTH$BITEST,
0000      ;         MTH$BJTEST, MTH$IIBSET, MTH$JIBSET, MTH$IIBCLR, MTH$JIBCLR.
0000      ;         JAW  21-May-1981
0000      ; 1-005 - Remove the routines added in edit 1-004 and place them in
0000      ;         FOR$BITOPS as FOR$ routines.  JAW 06-Jun-1981
```

```
    0000          .SBTTL  DECLARATIONS
    0000
    0000   ;
    0000   ; INCLUDE FILES:
    0000   ;        oerr.mar
    0000   ;
    0000
    0000   ;
    0000   ; EXTERNAL SYMBOLS:
    0000   ;        NONE
    0000   ;
    0000
    0000   ;
    0000   ; MACROS:
    0000   ;        NONE
    0000   ;
    0000
    0000   ;
    0000   ; PSECT DECLARATIONS:
00000000          .PSECT  _MTH$CODE       PIC, SHR, LONG, EXE, NOWRT
    0000
    0000   ;
    0000   ; EQUATED SYMBOLS:
    0000   ;        NONE
    0000   ;
    0000
    0000   ;
    0000   ; OWN STORAGE:
    0000   ;        NONE
    0000   ;
```

```
                             0000    98         .SBTTL  MTH$IIAND
                             0000    99
                             0000   100 ;++
                             0000   101 ; FUNCTIONAL DESCRIPTION:
                             0000   102 ;       Return the bitwise AND of two one-word arguments.
                             0000   103 ;
                             0000   104 ;
                             0000   105 ; CALLING SEQUENCE:
                             0000   106 ;       Bitwise_AND.ww.v = MTH$IIAND (arg1.rw.r, arg2.rw.r)
                             0000   107 ;
                             0000   108 ;
                             0000   109 ;
                             0000   110 ; INPUT PARAMETERS:
                             0000   111 ;       Both arguments are one-word values and are call-by-reference.
                             0000   112 ;
                             0000   113 ;
                             0000   114 ; IMPLICIT INPUTS:
                             0000   115 ;       NONE
                             0000   116 ;
                             0000   117 ; OUTPUT PARAMETERS:
                             0000   118 ;       NONE
                             0000   119 ;
                             0000   120 ; IMPLICIT OUTPUTS:
                             0000   121 ;       NONE
                             0000   122 ;
                             0000   123 ; COMPLETION CODES:
                             0000   124 ;       NONE
                             0000   125 ;
                             0000   126 ; SIDE EFFECTS:
                             0000   127 ;       NONE
                             0000   128 ;
                             0000   129 ;--
                             0000   130
                       0000  0000   131         .ENTRY  MTH$IIAND,      ^M<>
          50    04 BC    B2  0002   132         MCOMW   @4(AP), R0      ; R0 = complemented first arg
     50   08 BC    50    AB  0006   133         BICW3   R0, @8(AP), R0  ; R0 = ANDed args
                         04  000B   134         RET
```

```
                               000C          .SBTTL  MTH$JIAND
                               000C
                               000C    ;++
                               000C    ; FUNCTIONAL DESCRIPTION:
                               000C    ;         Return the bitwise AND of two longword arguments.
                               000C    ;
                               000C    ;
                               000C    ; CALLING SEQUENCE:
                               000C    ;         Bitwise_AND.wl.v = MTH$JIAND (arg1.rl.r, arg2.rl.r)
                               000C    ;
                               000C    ;
                               000C    ;
                               000C    ; INPUT PARAMETERS:
                               000C    ;         The two parameters are longword values and are call-by-reference.
                               000C    ;
                               000C    ;
                               000C    ; IMPLICIT INPUTS:
                               000C    ;         NONE
                               000C    ;
                               000C    ; OUTPUT PARAMETERS:
                               000C    ;         NONE
                               000C    ;
                               000C    ; IMPLICIT OUTPUTS:
                               000C    ;         NONE
                               000C    ;
                               000C    ; COMPLETION CODES:
                               000C    ;         NONE
                               000C    ;
                               000C    ; SIDE EFFECTS:
                               000C    ;         NONE
                               000C    ;
                               000C    ;--
                               000C
                               000C
                         0000  000C          .ENTRY  MTH$JIAND,         ^M<>
          50    04 BC      D2  000E          MCOML   @4(AP), R0         ; R0 = complemented first arg
   50     08 BC      50    CB  0012          BICL3   R0, @8(AP), R0     ; R0 = ANDed args
                           04  0017          RET
```

```
                    0018          .SBTTL  MTH$IIOR
                    0018
                    0018  ;++
                    0018  ; FUNCTIONAL DESCRIPTION:
                    0018  ;       Return the bitwise inclusive OR of two one-word arguments.
                    0018  ;
                    0018  ;
                    0018  ; CALLING SEQUENCE:
                    0018  ;       Bitwise_inclusive_OR.ww.v = MTH$IIOR (arg1.rw.r, arg2.rw.r)
                    0018  ;
                    0018  ;
                    0018  ;
                    0018  ; INPUT PARAMETERS:
                    0018  ;       The two parameters are one-word values and are call-by-reference.
                    0018  ;
                    0018  ;
                    0018  ; IMPLICIT INPUTS:
                    0018  ;       NONE
                    0018  ;
                    0018  ; OUTPUT PARAMETERS:
                    0018  ;       NONE
                    0018  ;
                    0018  ; IMPLICIT OUTPUTS:
                    0018  ;       NONE
                    0018  ;
                    0018  ; COMPLETION CODES:
                    0018  ;       NONE
                    0018  ;
                    0018  ; SIDE EFFECTS:
                    0018  ;       NONE
                    0018  ;
                    0018  ;--
                    0018
                    0018
               0000 0018          .ENTRY  MTH$IIOR,       ^M<>
50   08 BC   04 BC   A9 001A          BISW3   @4(AP), @8(AP), R0 ; R0 = ORed args
                 04 0020          RET
```

```
                         0021         .SBTTL  MTH$JIOR
                         0021
                         0021   ;++
                         0021   ; FUNCTIONAL DESCRIPTION:
                         0021   ;       Return the bitwise inclusive OR of two longword arguments.
                         0021   ;
                         0021   ;
                         0021   ; CALLING SEQUENCE:
                         0021   ;       Bitwise_inclusive_OR.wl.v = MTH$JIOR (arg1.rl.r, arg2.rl.r)
                         0021   ;
                         0021   ;
                         0021   ;
                         0021   ; INPUT PARAMETERS:
                         0021   ;       The two parameters are longword values and are call-by-reference.
                         0021   ;
                         0021   ;
                         0021   ; IMPLICIT INPUTS:
                         0021   ;       NONE
                         0021   ;
                         0021   ; OUTPUT PARAMETERS:
                         0021   ;       NONE
                         0021   ;
                         0021   ; IMPLICIT OUTPUTS:
                         0021   ;       NONE
                         0021   ;
                         0021   ; COMPLETION CODES:
                         0021   ;       NONE
                         0021   ;
                         0021   ; SIDE EFFECTS:
                         0021   ;       NONE
                         0021   ;
                         0021   ;--
                         0021
                         0021
                   0000  0021         .ENTRY  MTH$JIOR,       ^M<>
50    08 BC    04 BC  C9  0023         BISL3   @4(AP), @8(AP), R0 ; R0 = ORed args
                     04  0029         RET
```

```
                           002A   251          .SBTTL  MTH$IIEOR
                           002A   252
                           002A   253  ;++
                           002A   254  ; FUNCTIONAL DESCRIPTION:
                           002A   255  ;        Return the bitwise XOR of two one-word arguments.
                           002A   256  ;
                           002A   257  ;
                           002A   258  ; CALLING SEQUENCE:
                           002A   259  ;        Bitwise_exclusive_OR.ww.v = MTH$IIEOR (arg1.rw.r, arg2.rw.r)
                           002A   260  ;
                           002A   261  ;
                           002A   262  ;
                           002A   263  ; INPUT PARAMETERS:
                           002A   264  ;        The two parameters are one-word values and are call-by-reference.
                           002A   265  ;
                           002A   266  ;
                           002A   267  ; IMPLICIT INPUTS:
                           002A   268  ;        NONE
                           002A   269  ;
                           002A   270  ; OUTPUT PARAMETERS:
                           002A   271  ;        NONE
                           002A   272  ;
                           002A   273  ; IMPLICIT OUTPUTS:
                           002A   274  ;        NONE
                           002A   275  ;
                           002A   276  ; COMPLETION CODES:
                           002A   277  ;        NONE
                           002A   278  ;
                           002A   279  ; SIDE EFFECTS:
                           002A   280  ;        NONE
                           002A   281  ;
                           002A   282  ;--
                           002A   283
                           002A   284
                     0000  002A   285          .ENTRY  MTH$IIEOR,      ^M<>
  50   08 BC   04 BC   AD  002C   286          XORW3   @4(AP), @8(AP), R0 ; R0 = XORed args
                       04  0032   287          RET
```

```
                              0033          .SBTTL  MTH$JIEOR
                              0033
                              0033    ;++
                              0033    ; FUNCTIONAL DESCRIPTION:
                              0033    ;         Return the bitwise XOR of two longword arguments.
                              0033    ;
                              0033    ;
                              0033    ; CALLING SEQUENCE:
                              0033    ;         Bitwise_exclusive_or.wl.v = MTH$JIEOR (arg1.rl.r, arg2.rl.r)
                              0033    ;
                              0033    ;
                              0033    ;
                              0033    ; INPUT PARAMETERS:
                              0033    ;         The two arguments are longword values and are call-by-reference.
                              0033    ;
                              0033    ;
                              0033    ; IMPLICIT INPUTS:
                              0033    ;         NONE
                              0033    ;
                              0033    ; OUTPUT PARAMETERS:
                              0033    ;         NONE
                              0033    ;
                              0033    ; IMPLICIT OUTPUTS:
                              0033    ;         NONE
                              0033    ;
                              0033    ; COMPLETION CODES:
                              0033    ;         NONE
                              0033    ;
                              0033    ; SIDE EFFECTS:
                              0033    ;         NONE
                              0033    ;
                              0033    ;--
                              0033
                              0033
                        0000  0033            .ENTRY  MTH$JIEOR,      ^M<>
   50   08 BC   04 BC     CD  0035            XORL3   @4(AP), @8(AP), R0 ; R0 = XORed args
                          04  003B            RET
```

```
                    003C   327          .SBTTL  MTH$INOT
                    003C   328
                    003C   329  ;++
                    003C   330  ; FUNCTIONAL DESCRIPTION:
                    003C   331  ;       Return the bitwise NOT of a one-word argument.
                    003C   332  ;
                    003C   333  ;
                    003C   334  ; CALLING SEQUENCE:
                    003C   335  ;       Bitwise_complement.ww.v = MTH$INOT (arg.rw.r)
                    003C   336  ;
                    003C   337  ;
                    003C   338  ;
                    003C   339  ; INPUT PARAMETERS:
                    003C   340  ;       The one parameter is a one-word value and is call-by-reference.
                    003C   341  ;
                    003C   342  ;
                    003C   343  ; IMPLICIT INPUTS:
                    003C   344  ;       NONE
                    003C   345  ;
                    003C   346  ; OUTPUT PARAMETERS:
                    003C   347  ;       NONE
                    003C   348  ;
                    003C   349  ; IMPLICIT OUTPUTS:
                    003C   350  ;       NONE
                    003C   351  ;
                    003C   352  ; COMPLETION CODES:
                    003C   353  ;       NONE
                    003C   354  ;
                    003C   355  ; SIDE EFFECTS:
                    003C   356  ;       NONE
                    003C   357  ;
                    003C   358  ;--
                    003C   359
                    003C   360
               0000 003C   361          .ENTRY  MTH$INOT,       ^M<>
  50    04 BC    B2 003E   362          MCOMW   @4(AP), R0      ; R0 = complemented arg
                 04 0042   363          RET
```

```
                    0043          .SBTTL  MTH$JNOT
                    0043
                    0043  ;++
                    0043  ; FUNCTIONAL DESCRIPTION:
                    0043          Return the bitwise NOT of a longword argument.
                    0043  ;
                    0043  ;
                    0043  ; CALLING SEQUENCE:
                    0043          Bitwise_complement.wl.v=MTH$JNOT (arg.rl.r)
                    0043  ;
                    0043  ;
                    0043  ;
                    0043  ; INPUT PARAMETERS:
                    0043          The one parameter is a longword value and is call-by-reference.
                    0043  ;
                    0043  ;
                    0043  ; IMPLICIT INPUTS:
                    0043          NONE
                    0043  ;
                    0043  ; OUTPUT PARAMETERS:
                    0043          NONE
                    0043  ;
                    0043  ; IMPLICIT OUTPUTS:
                    0043          NONE
                    0043  ;
                    0043  ; COMPLETION CODES:
                    0043          NONE
                    0043  ;
                    0043  ; SIDE EFFECTS:
                    0043          NONE
                    0043  ;
                    0043  ;--
                    0043
                    0043
              0000  0043          .ENTRY  MTH$JNOT,       ^M<>
   50   04 BC   D2  0045          MCOML   @4(AP), R0      ; R0 = complemented arg
                04  0049          RET
```

```
                         004A          .SBTTL  MTH$IISHFT
                         004A
                         004A   ;++
                         004A   ; FUNCTIONAL DESCRIPTION:
                         004A   ;       Return the result of shifting arg1 by arg2 places.
                         004A   ;
                         004A   ;
                         004A   ; CALLING SEQUENCE:
                         004A   ;       Bitwise_shift.ww.v = MTH$IISHFT (arg1.rwu.r, shiftcount.rw.r)
                         004A   ;
                         004A   ;
                         004A   ;
                         004A   ;
                         004A   ; INPUT PARAMETERS:
                         004A   ;       Both arguments are one-word values and are call-by-reference.
                         004A   ;
                         004A   ;
                         004A   ;
                         004A   ; IMPLICIT INPUTS:
                         004A   ;       NONE
                         004A   ;
                         004A   ; OUTPUT PARAMETERS:
                         004A   ;       NONE
                         004A   ;
                         004A   ; IMPLICIT OUTPUTS:
                         004A   ;       NONE
                         004A   ;
                         004A   ; COMPLETION CODES:
                         004A   ;       NONE
                         004A   ;
                         004A   ; SIDE EFFECTS:
                         004A   ;       NONE
                         004A   ;
                         004A   ;--
                         004A
                         004A
                   0000  004A          .ENTRY  MTH$IISHFT,      ^M<>
       50   04 BC   3C   004C          MOVZWL  @4(AP), R0       ; R0 = zero-extended arg
  50   50   08 BC   78   0050          ASHL    @8(AP), R0, R0   ; so right shift brings in zeroes
                    04   0055          RET
```

```
                              0056          .SBTTL  MTH$JISHFT
                              0056
                              0056  ;++
                              0056  ; FUNCTIONAL DESCRIPTION:
                              0056  ;
                              0056  ;       Return the result of shifting arg1 by arg2 places
                              0056  ; CALLING SEQUENCE:
                              0056  ;       Bitwise-shift.wl.v = MTH$JISHFT (arg1.rlu.r, shiftcount.rl.r)
                              0056  ;
                              0056  ;
                              0056  ; INPUT PARAMETERS:
                              0056  ;       Both arguments are longword values and are call-by-references.
                              0056  ;
                              0056  ; IMPLICIT INPUTS:
                              0056  ;       NONE
                              0056  ;
                              0056  ; OUTPUT PARAMETERS:
                              0056  ;       NONE
                              0056  ;
                              0056  ; IMPLICIT OUTPUTS:
                              0056  ;       NONE
                              0056  ;
                              0056  ; COMPLETION CODES:
                              0056  ;       NONE
                              0056  ;
                              0056  ; SIDE EFFECTS:
                              0056  ;       NONE
                              0056  ;
                              0056  ;--
                              0056
                              0056
                         0000 0056          .ENTRY  MTH$JISHFT,      ^M<>
         50    04 BC     D0   0058          MOVL    @4(AP), R0       ; R0 = arg to shift
                   51    D4   005C          CLRL    R1               ; R1 = 0 so right shift brings in zeroes
  50     50    08 BC     79   005E          ASHQ    @8(AP), R0, R0   ; shift R0 by arg2 bits
                         04   0063          RET
                              0064
                              0064          .END
```

```
MTH$IIAND          00000000 RG      01
MTH$IIEOR          0000002A RG      01
MTH$IIOR           00000018 RG      01
MTH$IISHFT         0000004A RG      01
MTH$INOT           0000003C RG      01
MTH$JIAND          0000000C RG      01
MTH$JIEOR          00000033 RG      01
MTH$JIOR           00000021 RG      01
MTH$JISHFT         00000056 RG      01
MTH$JNOT           00000043 RG      01

                                                  +----------------+
                                                  ! Psect synopsis !
                                                  +----------------+

PSECT name                        Allocation           PSECT No.  Attributes
----------                        ----------           ---------  ----------
.  ABS  .                         00000000  (     0.)  00 (  0.)  NOPIC   USR   CON   ABS   LCL NOSHR NOEXE NORD  NOWRT NOVEC BYTE
_MTH$CODE                         00000064  (   100.)  01 (  1.)    PIC   USR   CON   REL   LCL   SHR   EXE   RD  NOWRT NOVEC LONG

                                               +------------------------+
                                               ! Performance indicators !
                                               +------------------------+

Phase                        Page faults    CPU Time       Elapsed Time
-----                        -----------    --------       ------------
Initialization                        37    00:00:00.06    00:00:01.93
Command processing                   138    00:00:00.49    00:00:02.89
Pass 1                                76    00:00:00.77    00:00:04.10
Symbol table sort                      0    00:00:00.00    00:00:00.00
Pass 2                                83    00:00:00.86    00:00:04.34
Symbol table output                    2    00:00:00.01    00:00:00.02
Psect synopsis output                  2    00:00:00.01    00:00:00.01
Cross-reference output                 0    00:00:00.00    00:00:00.00
Assembler run totals                 340    00:00:02.21    00:00:13.29

The working set limit was 750 pages.
4700 bytes (10 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 10 non-local and 0 local symbols.
479 source lines were read in Pass 1, producing 37 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.

                                              +--------------------------+
                                              ! Macro library statistics !
                                              +--------------------------+

Macro library name                             Macros defined
------------------                             --------------
_$255$DUA28:[SYSLIB]STARLET.MLB;2                          0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:MTHBITOPS/OBJ=OBJ$:MTHBITOPS MSRC$:MTHBITOPS/UPDATE=(ENH$:MTHBITOPS)
```

0257 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY